Machine Learning BCS602

Scheme & solution

May 2026

Question: - 1a)

	Module – 1	M	L	C
a.	Define Machine Learning. Explain different types of machine learning.	10	L2	CO1

1a)

Scheme:-

Definition: - 2 Marks

Types: - 4*2=8 Marks

Definition Machine Learning (ML) is a branch of Artificial Intelligence that enables computers to learn from data and improve their performance without being explicitly programmed. It uses algorithms to identify patterns in data and make predictions or decisions automatically.

Example: Email spam filtering, recommendation systems, image recognition, and speech recognition.

Types of Machine Learning

1. Supervised Learning

Supervised learning uses **labelled data**, where both input and output are known. The algorithm learns the relationship between inputs and outputs and predicts results for new data.

Applications:

- Spam email detection
- House price prediction
- Disease diagnosis

Algorithms: Linear Regression, Decision Trees, SVM

2. Unsupervised Learning

Unsupervised learning uses **unlabelled data**. The algorithm identifies hidden patterns, structures, or groups within the data.

Applications:

- Customer segmentation
- Market basket analysis
- Data clustering

Algorithms: K-Means, Hierarchical Clustering, PCA

3. Semi-Supervised Learning

Semi-supervised learning uses a combination of **small labelled data** and **large unlabelled data**. It improves learning accuracy while reducing the effort of data labelling.

Applications:

- Image classification
- Speech recognition
- Text classification

Algorithms: Self-Training, Label Propagation

4. Reinforcement Learning

Reinforcement learning is based on **trial and error**. An agent interacts with an environment and receives rewards or penalties for its actions. The goal is to maximize rewards over time.

Applications:

- Robotics
- Self-driving cars
- Game playing

Algorithms: Q-Learning, SARSA, Deep Q Networks

Comparison of Types of ML

Type	Data Used	Purpose
Supervised Learning	Labelled Data	Prediction and Classification
Unsupervised Learning	Unlabelled Data	Pattern Discovery
Semi-Supervised Learning	Labelled + Unlabelled Data	Improved Learning Accuracy
Reinforcement Learning	Reward-Based Data	Optimal Decision Making

Machine Learning allows computers to learn from experience and data. The four major types—**Supervised Learning, Unsupervised Learning, Semi-Supervised Learning, and Reinforcement Learning**—are widely used in solving real-world problems such as prediction, classification, clustering, and intelligent decision-making. This makes ML a key technology in modern AI applications.

Question 1b)

b.	Consider the following set : S = {12, 14, 19, 22, 24, 26, 28, 31, 32}. Apply various binning techniques and show the result.	**6**	**L3**	**CO1**

1b)

Scheme:-

Binning Techniques: - 2*3=6 Marks

Given:

$$S = \{12,14,19,22,24,26,28,31,32\}$$

Apply **binning techniques** with **equal-depth bins** (3 bins, each containing 3 values).

Step 1: Create Bins

Bin	Values
B1	**12, 14, 19**
B2	**22, 24, 26**
B3	**28, 31, 32**

1. Binning by Means

Calculate the mean of each bin and replace all values in the bin with the mean.

Bin 1:

$$\text{Mean} = \frac{12 + 14 + 19}{3} = 15$$

→ (15, 15, 15)

Bin 2:

$$\text{Mean} = \frac{22 + 24 + 26}{3} = 24$$

→ (24, 24, 24)

Bin 3:

$$\text{Mean} = \frac{28 + 31 + 32}{3} = 30.33 \approx 30$$

→ (30, 30, 30)

Result (Binning by Means)

$$\{15, 15, 15, 24, 24, 24, 30, 30, 30\}$$

2. Binning by Medians

Replace each value in a bin by the median of that bin.

Bin 1:

Median = 14

→ (14, 14, 14)

Bin 2:

Median = 24

→ (24, 24, 24)

Bin 3:

Median = 31

→ (31, 31, 31)

Result (Binning by Medians)

$$\{14, 14, 14, 24, 24, 24, 31, 31, 31\}$$

3. Binning by Boundaries

Replace each value by the closest boundary (minimum or maximum value of the bin).

Bin 1: (12, 14, 19)

Boundaries = 12 and 19

- 12 → 12
- 14 → 12 (closer to 12)
- 19 → 19

Result: (12, 12, 19)

Bin 2: (22, 24, 26)

Boundaries = 22 and 26

- 22 → 22
- 24 → 22 (or 26; choose lower boundary when equidistant)
- 26 → 26

Result: (22, 22, 26)

Bin 3: (28, 31, 32)

Boundaries = 28 and 32

- 28 → 28
- 31 → 32 (closer to 32)
- 32 → 32

Result: (28, 32, 32)

Result (Binning by Boundaries)

$$\{12, 12, 19, 22, 22, 26, 28, 32, 32\}$$

Final Answer

Technique	Result
Binning by Means	{15, 15, 15, 24, 24, 24, 30, 30, 30}
Binning by Medians	{14, 14, 14, 24, 24, 24, 31, 31, 31}
Binning by Boundaries	{12, 12, 19, 22, 22, 26, 28, 32, 32}

Question 1C)

c.	Differentiate between the four types of attributes used in data analysis. Give suitable examples.	4	L2	CO1

1C)

Scheme: -

Types of attributes: -1*4=4 Marks

In data analysis, attributes (features) are classified into four main types based on the nature of the data they represent.

Attribute Type	Description	Characteristics	Examples
Nominal Attribute	Represents categories or names without any order.	Data can only be classified or grouped. Arithmetic operations are not possible.	Gender (Male/Female), Color (Red, Blue, Green), Blood Group (A, B, AB, O)
Ordinal Attribute	Represents categories with a meaningful order or ranking.	Ordering is possible, but differences between values cannot be measured.	Grades (A, B, C, D), Customer Satisfaction (Poor, Fair, Good, Excellent), Rank (1st, 2nd, 3rd)
Interval Attribute	Numeric data with equal intervals between values but no true zero point.	Addition and subtraction are meaningful; ratios are not.	Temperature in Celsius or Fahrenheit, Calendar Years
Ratio Attribute	Numeric data with equal intervals and a true zero point.	All arithmetic operations including multiplication and division are meaningful.	Height, Weight, Age, Salary, Distance

Examples

1. **Nominal:** Student Department = CSE, ISE, ECE
2. **Ordinal:** Performance Rating = Excellent, Good, Average, Poor
3. **Interval:** Temperature = 20°C, 30°C, 40°C
4. **Ratio:** Weight = 50 kg, 100 kg (100 kg is twice 50 kg)

The four types of attributes used in data analysis are **Nominal, Ordinal, Interval, and Ratio**. Understanding these attribute types helps in selecting appropriate statistical and data mining techniques for analysis. This classification is essential for effective data preprocessing and interpretation.

Question 2a)

Q.2	a.	Explain machine learning process model with neat diagram and list any two applications of machine learning.	10	L2	CO1

2a)

Scheme: -

Diagram: - 2 Marks

Phase wise explanation: - 6 Marks

Application (Any 2): - 2 Marks

Machine Learning (ML) is a branch of Artificial Intelligence that enables computers to learn from data and improve their performance without being explicitly programmed. The Machine Learning Process Model consists of a sequence of steps that transform raw data into a trained model capable of making predictions or decisions.

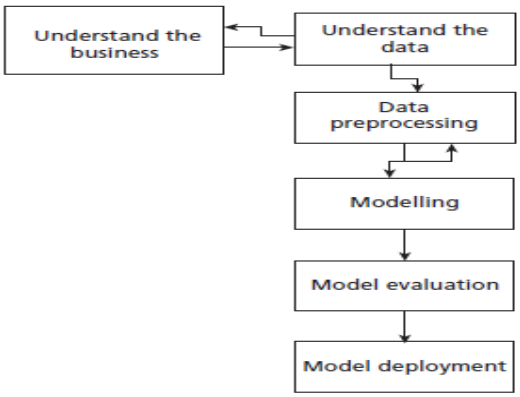


Figure 1.11: A Machine Learning/Data Mining Process

The diagram represents the **CRISP-DM (Cross-Industry Standard Process for Data Mining)** methodology, which is a widely used machine learning and data mining process model. It consists of six phases that help in developing an effective machine learning solution.

1. Understand the Business

- This is the first step where the problem to be solved is clearly identified.
- Business objectives, requirements, constraints, and success criteria are defined.
- The goal is to understand what the organization wants to achieve through machine learning.

Example: A bank wants to predict whether a customer will default on a loan.

2. Understand the Data

- Relevant data is collected and explored.
- Data quality, structure, patterns, and relationships are analyzed.
- Any missing values, inconsistencies, or anomalies are identified.

Example: Collecting customer income, age, credit history, and loan details.

3. Data Preprocessing

- Raw data is cleaned and transformed into a suitable format for analysis.
- Missing values are handled, duplicate records are removed, and data is normalized.
- Feature selection and feature engineering may also be performed.

Example: Filling missing income values and converting categorical data into numerical form.

4. Modelling

- Appropriate machine learning algorithms are selected and trained on the prepared data.
- Different models may be built and compared to identify the best-performing one.

Example: Applying Decision Tree, Random Forest, or Logistic Regression algorithms.

5. Model Evaluation

- The trained model is tested using validation or test data.
- Performance metrics such as Accuracy, Precision, Recall, and F1-score are calculated.
- The model is checked to ensure it satisfies business objectives.

Example: Evaluating how accurately the model predicts loan defaults.

6. Model Deployment

- Once the model performs satisfactorily, it is deployed in a real-world environment.
- The model is used to make predictions on new data and support decision-making.
- Continuous monitoring and maintenance are performed to ensure effectiveness.

Example: Deploying the loan approval model in the bank's online loan processing system.

Question 2b)

b.	Consider the set : V = {88, 90, 92, 94}. Apply min-max procedure and map the marks to a new range 0 – 1.	**6**	**L3**	**CO1**

2b)

Scheme: -

Min-Max Procedure: - 2 Marks (Formula)

Calculation: - 2 Marks

Apply Technique: -2 Marks

Given:

$$V = \{88,90,92,94\}$$

Apply **Min-Max Normalization** to map the values to the range **[0,1]**.

Formula

$$v' = \frac{v - \min(V)}{\max(V) - \min(V)}$$

where:

- $\min(V) = 88$
- $\max(V) = 94$

$$v' = \frac{v - 88}{94 - 88} = \frac{v - 88}{6}$$

Calculations

Original Value (v)	Normalized Value $(v - 88)/6$
88	$\frac{88-88}{6} = 0$
90	$\frac{90-88}{6} = \frac{2}{6} = 0.333$
92	$\frac{92-88}{6} = \frac{4}{6} = 0.667$
94	$\frac{94-88}{6} = 1$

Therefore, after applying Min-Max normalization to the range [0,1], the normalized values are:

$$\boxed{\{0,\ 0.333,\ 0.667,\ 1\}}$$

Question 2c)

c.	Explain any four data visualization techniques used for univariate data analysis.	4	L2	CO1

2C)

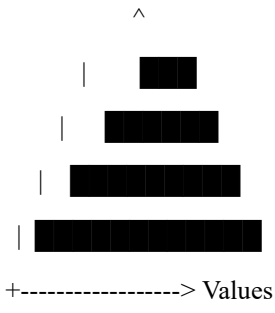
Scheme: -

Techniques: -1*4=4 Marks

1. Histogram

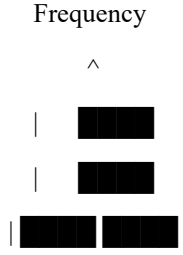
- Displays the frequency distribution of continuous data using adjacent bars.
- Helps understand the shape and spread of data.

Frequency



2. Bar Chart

- Represents categorical data using rectangular bars.
- Used to compare frequencies of different categories.

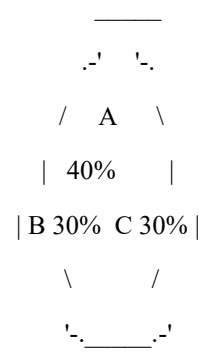


+----------> Categories

A B C

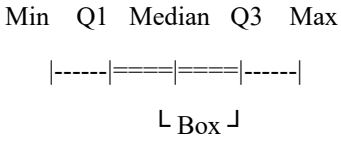
3. Pie Chart

- Shows the proportion or percentage of each category as slices of a circle.
- Useful for displaying part-to-whole relationships.



4. Box Plot (Box-and-Whisker Plot)

- Summarizes data using minimum, maximum, median, and quartiles.
- Helps identify data spread and outliers.



Question 3a)

Q.3	**a.**	Demonstrate Find-S algorithm for finding maximally specific hypothesis on the given dataset in Table 3(a).					**10**	**L3**	**CO2**

Table 3(a)

Origin	Manufacture	Color	Year	Type	Class
Japan	Honda	Blue	1980	Economy	Yes
Japan	Toyota	Green	1970	Sport	No
Japan	Toyota	Blue	1990	Economy	Yes
USA	Audi	Red	1980	Economy	No
Japan	Honda	White	1980	Economy	Yes
Japan	Toyota	Green	1980	Economy	Yes
Japan	Honda	Red	1980	Economy	No

3a)

Scheme:-

Writing the Find-S Algorithm / Initial Hypothesis (h_0): -1 Mark

Processing 1st to Positive Example correctly: - 1+2+2+2=7 Mark

Final Hypothesis obtained correctly: -1 Mark

Interpretation / Conclusion: -1 Mark

Step 1: Initialize Hypothesis

Find-S starts with the most specific hypothesis:

$$h_0 = \langle \phi, \phi, \phi, \phi, \phi \rangle$$

Step 2: Consider First Positive Example

E1 = (Japan, Honda, Blue, 1980, Economy) → Yes

Replace all φ with attribute values:

$$h_1 = \langle \text{Japan, Honda, Blue, 1980, Economy} \rangle$$

Step 3: Ignore Negative Example

E2 = (Japan, Toyota, Green, 1970, Sport) → No

Negative examples are ignored in Find-S.

$$h_2 = \langle \text{Japan, Honda, Blue, 1980, Economy} \rangle$$

Step 4: Process Positive Example

E3 = (Japan, Toyota, Blue, 1990, Economy) → Yes

Compare with h_2:

Attribute	**h₂**	**E3**	**Action**
Origin	Japan	Japan	Same
Manufacture	Honda	Toyota	?
Color	Blue	Blue	Same
Year	1980	1990	?
Type	Economy	Economy	Same

h3=⟨Japan, ?, Blue, ?, Economy⟩

Step 5: Ignore Negative Example

E4 = (USA, Audi, Red, 1980, Economy) → No

Ignored.

$$h_4 = \langle Japan, ?, Blue, ?, Economy \rangle$$

Step 6: Process Positive Example

E5 = (Japan, Honda, White, 1980, Economy) → Yes

Compare with h_4:

Attribute	**h₄**	**E5**	**Action**
Origin	Japan	Japan	Same
Manufacture	?	Honda	Remains ?

Attribute	h_4	E5	Action
Color	Blue	White	?
Year	?	1980	Remains ?
Type	Economy	Economy	Same

$$h5=\langle Japan, ?, ?, ?, Economy\rangle$$

Step 7: Process Positive Example

E6 = (Japan, Toyota, Green, 1980, Economy) → Yes

Compare with h_5:

Attribute	h_5	E6	Action
Origin	Japan	Japan	Same
Manufacture	?	Toyota	Remains ?
Color	?	Green	Remains ?
Year	?	1980	Remains ?
Type	Economy	Economy	Same

$$h6=\langle Japan, ?, ?, ?, Economy\rangle$$

Step 8: Ignore Negative Example

E7 = (Japan, Honda, Red, 1980, Economy) → No

Ignored.

$$h_7 = \langle Japan, ?, ?, ?, Economy\rangle$$

Final Hypothesis

$$\boxed{h = \langle Japan, ?, ?, ?, Economy\rangle}$$

Interpretation

A car is classified as **Yes** if:

- **Origin = Japan**
- **Type = Economy**
- Manufacture, Color, and Year can be any value

Question 3b)

b.	Write steps involved in PCA (Principal Component Analysis) algorithm.	5	L2	CO2

3b)

Scheme:-

Definition: - 1 Marks

Steps: - 3 Marks

Diagram: - 1 Marks

PCA is a dimensionality reduction technique used to reduce the number of features in a dataset while retaining maximum information (variance).

Steps Involved in PCA

1. **Standardize the Data**
    - Normalize the dataset so that all features have the same scale.
2. **Compute the Covariance Matrix**
    - Determine the relationships among the features.
3. **Calculate Eigenvalues and Eigenvectors**
    - Find the principal components and their importance.
4. **Select Principal Components**
    - Choose the components with the highest eigenvalues.
5. **Transform the Data**
    - Project the original data onto the selected principal components to obtain a reduced-dimensional dataset.

Dataset

↓

Standardization

↓

Covariance Matrix

↓

Eigenvalues & Eigenvectors

↓

Select Principal Components

↓

Transformed Data

Question 3c)

c.	Explain any five metrics used to evaluate the performance of machine learning model.	5	L2	CO2

3c)

Scheme:-

Metrics: -1*5=5 Marks

1. Accuracy

- Accuracy measures the proportion of correctly predicted instances to the total instances.
- It is suitable when the dataset is balanced.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

2. Precision

- Precision measures how many of the positively predicted instances are actually positive.
- It is important when false positives are costly.

$$Precision = \frac{TP}{TP + FP}$$

3. Recall (Sensitivity)

- Recall measures how many actual positive instances are correctly identified.
- It is useful when false negatives are costly.

$$\boldsymbol{Recall = \frac{TP}{TP + FN}}$$

4. F1-Score

- F1-Score is the harmonic mean of Precision and Recall.
- It provides a balance between Precision and Recall.

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

5. Mean Squared Error (MSE)

- MSE is commonly used for regression models.
- It measures the average squared difference between actual and predicted values.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Question 4a)

OR

Q.4	a.	Apply Candidate Elimination Algorithm (CEA) on the following dataset in Table 4(a) and determine the final version space.	10	L3	CO2

Table 4(a)

Sl. No. Example	Sky	Air Temp	Humidity	Wind	Water	Forecast	Enjoy sport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	Strong	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Warm	Change	No
4	Sunny	Warm	High	Strong	Cool	Change	Yes

4a)

Scheme:-

Initialization → 1 Mark

Processing Positive Examples → 3 Marks

Processing Negative Example → 3 Marks

Final Boundaries (S and G) → 3 Marks

Step 1: Initialize

Specific Boundary (S)

Most specific hypothesis:

$$S_0 = \langle \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset \rangle$$

General Boundary (G)

Most general hypothesis:

$$\boldsymbol{G_0} = \langle ?, ?, ?, ?, ?, ? \rangle$$

Step 2: Process Example E1 (Positive)

E1 = ⟨Sunny, Warm, Normal, Strong, Warm, Same⟩

Update S to cover E1:

$$S_1 = \langle Sunny, Warm, Normal, Strong, Warm, Same \rangle$$

G remains unchanged:

$$G_1 = \langle ?, ?, ?, ?, ?, ? \rangle$$

Step 3: Process Example E2 (Positive)

E2 = ⟨Sunny, Warm, High, Strong, Warm, Same⟩

Compare with S_1:

Attribute	S_1	E2	Result
Sky	**Sunny**	**Sunny**	**Same**
Air Temp	**Warm**	**Warm**	**Same**
Humidity	**Normal**	**High**	**Different → ?**
Wind	**Strong**	**Strong**	**Same**
Water	**Warm**	**Warm**	**Same**
Forecast	**Same**	**Same**	**Same**

Therefore,

$$\boldsymbol{S_2 = \langle Sunny, Warm, ?, Strong, Warm, Same \rangle}$$

G remains:

$$\boldsymbol{G_2} = \langle ?, ?, ?, ?, ?, ? \rangle$$

Step 4: Process Example E3 (Negative)

E3 = ⟨Rainy, Cold, High, Strong, Warm, Change⟩

Since it is a negative example, specialize G so that it excludes E3 but remains consistent with S_2.

Possible specializations:

1. ⟨Sunny, ?, ?, ?, ?, ?⟩
2. ⟨?, Warm, ?, ?, ?, ?⟩
3. ⟨?, ?, ?, ?, ?, Same⟩

(Remove hypotheses inconsistent with S_2.)

Thus,

$$G_3 = \{\langle Sunny, ?, ?, ?, ?, ? \rangle, \langle ?, Warm, ?, ?, ?, ? \rangle, \langle ?, ?, ?, ?, ?, Same \rangle\}$$

S remains:

$$S_3 = \langle Sunny, Warm, ?, Strong, Warm, Same \rangle$$

Step 5: Process Example E4 (Positive)

E4 = ⟨Sunny, Warm, High, Strong, Cool, Change⟩

Compare with S_3:

Attribute	**S_3**	**E4**	**Result**
Sky	**Sunny**	**Sunny**	**Same**
Air Temp	**Warm**	**Warm**	**Same**
Humidity	**?**	**High**	**?**
Wind	**Strong**	**Strong**	**Same**
Water	**Warm**	**Cool**	**Different → ?**
Forecast	**Same**	**Change**	**Different → ?**

Therefore,

$$\boldsymbol{S_4 = \langle Sunny, Warm, ?, Strong, ?, ? \rangle}$$

Now remove from G any hypothesis that does not cover E4.

- **⟨Sunny, ?, ?, ?, ?, ?⟩ ✓ Covers E4**
- **⟨?, Warm, ?, ?, ?, ?⟩ ✓ Covers E4**
- **⟨?, ?, ?, ?, ?, Same⟩ ✗ Does not cover E4**

Remove the third hypothesis.

Hence,

$$G_4 = \{\langle \textbf{Sunny}, ?, ?, ?, ?, ?\rangle, \langle ?, \textbf{Warm}, ?, ?, ?, ?\rangle\}$$

Final Version Space

Specific Boundary (S)

$$\boxed{S = \langle \textbf{Sunny}, \textbf{Warm}, ?, \textbf{Strong}, ?, ?\rangle}$$

General Boundary (G)

$$\boxed{G = \{\langle \textbf{Sunny}, ?, ?, ?, ?, ?\rangle, \langle ?, \textbf{Warm}, ?, ?, ?, ?\rangle\}}$$

Final Answer

- **Specific Boundary:**

$$\boxed{\langle \textbf{Sunny}, \textbf{Warm}, ?, \textbf{Strong}, ?, ?\rangle}$$

- **General Boundary:**

$$\boxed{\{\langle \textbf{Sunny}, ?, ?, ?, ?, ?\rangle, \langle ?, \textbf{Warm}, ?, ?, ?, ?\rangle\}}$$

Question 4b)

b.	Explain Bivariate Statistics. Describe covariance and correlation with formulas and interpretation.	**5**	**L2**	**CO2**

4b)

Scheme:-

Definition: - 1*3=3Marks

Formulas: -1*2=2 Marks

Bivariate Statistics is the statistical analysis of two variables simultaneously to determine the relationship or association between them.

Example: Study Hours and Exam Marks.

Covariance – Definition and Formula (2 Marks)

Covariance measures the **direction of the relationship** between two variables.

Formula

$$Cov(X,Y) = \frac{\sum(X_i - \bar{X})(Y_i - \bar{Y})}{n-1}$$

Interpretation

- **Cov(X,Y) > 0:** Positive relationship (both variables increase together).
- **Cov(X,Y) < 0:** Negative relationship (one variable increases while the other decreases).
- **Cov(X,Y) = 0:** No linear relationship.

Correlation – Definition, Formula and Interpretation (2 Marks)

Correlation measures the **strength and direction** of the relationship between two variables.

Formula

$$r = \frac{Cov(X,Y)}{\sigma_X \sigma_Y}$$

Interpretation

- **r = +1:** Perfect positive correlation.
- **r = 0:** No correlation.
- **r = -1:** Perfect negative correlation.

Question 4c)

c.	Explain Re-sampling methods for model evaluation.	**5**	**L2**	**CO2**

4c)

Scheme:-

Definition: - 2 Marks

Methods & Explanation=1+3=4 Marks

Resampling methods are statistical techniques used to evaluate the performance of a machine learning model by repeatedly drawing samples from the available dataset. They help estimate how well a model will perform on unseen data.

Methods

1.Hold-Out Method

- The dataset is divided into **training** and **testing** sets.
- The model is trained on the training set and evaluated on the testing set.
- Common split: **80% training and 20% testing**.

Advantage: Simple and fast.

2. **k-Fold Cross Validation (1.5 Marks)**

- The dataset is divided into **k equal folds**.
- The model is trained on **k−1 folds** and tested on the remaining fold.
- This process is repeated **k times**, with each fold used once as the test set.
- The average performance is taken as the final result.

Example: 5-Fold or 10-Fold Cross Validation.

Advantage: Provides a more reliable evaluation.

3. **Bootstrap Method (1.5 Marks)**

- Random samples are drawn **with replacement** from the original dataset.
- Some observations may appear multiple times, while others may not appear.
- The model is trained on the bootstrap sample and tested on the remaining data.

Advantage: Useful when the dataset is small.

Question 5a)

Q.5	a.	Using the K-Nearest Neighbors (KNN) algorithm with K = 3 and Euclidean distance, classify the result of student having (CGPA = 6.1, Assessment = 40, Project submitted = 5) based on dataset in Table 5(a).	10	L3	CO3

Table 5(a)

Sl. No.	CGPA	Assessment	Project Submitted	Result
1	9.2	85	8	Pass
2	8	80	7	Pass
3	8.5	81	8	Pass
4	6	45	5	Fail
5	6.5	50	4	Fail
6	8.2	72	7	Pass
7	5.8	38	5	Fail
8	8.9	91	9	Pass

5a)

Scheme: -

Euclidean Distance Formula: -2Mark

Distance Calculations for All Students: -3 Marks

Identification of 3 Nearest Neighbours: -1 Mark

Majority Voting and Classification: -2 Mark

Conclusion/Prediction: -1 Mark

Table 4.3: Euclidean Distance

S.No.	CGPA	Assessment	Project Submitted	Result	Euclidean Distance
1.	9.2	85	8	Pass	$\sqrt{(9.2-6.1)^2+(85-40)^2+(8-5)^2}$ $=45.2063$
2.	8	80	7	Pass	$\sqrt{(8-6.1)^2+(80-40)^2+(7-5)^2}$ $=40.09501$
3.	8.5	81	8	Pass	$\sqrt{(8.5-6.1)^2+(81-40)^2+(8-5)^2}$ $=41.17961$
4.	6	45	5	Fail	$\sqrt{(6-6.1)^2+(45-40)^2+(5-5)^2}$ $=5.001$
5.	6.5	50	4	Fail	$\sqrt{(6.5-6.1)^2+(50-40)^2+(4-5)^2}$ $=10.05783$
6.	8.2	72	7	Pass	$\sqrt{(8.2-6.1)^2+(72-40)^2+(7-5)^2}$ $=32.13114$
7.	5.8	38	5	Fail	$\sqrt{(5.8-6.1)^2+(38-40)^2+(5-5)^2}$ $=2.022375$
8.	8.9	91	9	Pass	$\sqrt{(8.9-6.1)^2+(91-40)^2+(9-5)^2}$ $=51.23319$

Step 2: Sort the distances in the ascending order and select the first 3 nearest training data instances to the test instance. The selected nearest neighbors are shown in Table 4.4.

Table 4.4: Nearest Neighbors

Instance	Euclidean Distance	Class
4	5.001	Fail
5	10.05783	Fail
7	2.022375	Fail

Here, we take the 3 nearest neighbors as instances 4, 5 and 7 with smallest distances.

Step 3: Predict the class of the test instance by majority voting.

The class for the test instance is predicted as 'Fail'.

Question 5b)

b.	List and explain different types of regression methods used in machine learning.	**4**	**L2**	**CO3**

5b)

Scheme:-

Definition of Regression: -1 Mark

Any Three Regression Methods (1 Mark each): -3 Marks

Regression is a supervised machine learning technique used to predict a continuous numerical value by finding the relationship between dependent and independent variables.

Types of Regression Methods

1. Linear Regression

- Establishes a linear relationship between one independent variable and one dependent variable.
- Used for predicting continuous values such as marks or sales.

2. Multiple Linear Regression

- Uses two or more independent variables to predict a dependent variable.
- Example: Predicting salary based on experience and education.

3. Polynomial Regression

- Used when the relationship between variables is nonlinear.
- Fits a curved line to the data for better prediction.

Question 5c)

c.	Explain the procedure to construct a decision tree using ID_3 algorithm.	**6**	**L2**	**CO3**

5c)

Scheme:

Definition of ID3 Algorithm: -2 Marks

Entropy and Information Gain: -4 Marks

ID3 Algorithm

ID3 (Iterative Dichotomiser 3) is a decision tree algorithm used for classification. It constructs a decision tree by selecting the attribute with the highest Information Gain at each node.

Entropy and Information Gain (1 Mark)

Entropy measures the impurity or uncertainty in a dataset.

$$Entropy(S) = -\sum p_i \log_2(p_i)$$

Information Gain measures the reduction in entropy after splitting the dataset.

$$\boldsymbol{IG(S,A) = Entropy(S) - \sum \frac{|\,S_v\,|}{|\,S\,|} Entropy(S_v)}$$

Procedure of ID3 Algorithm

Step 1: Calculate Entropy (1 Mark)

Compute the entropy of the entire dataset to measure its impurity.

Step 2: Calculate Information Gain (1 Mark)

Calculate the Information Gain for each attribute and determine how effectively it separates the data.

Step 3: Select the Best Attribute (1 Mark)

Choose the attribute with the highest Information Gain as the root node of the decision tree.

Step 4: Create Node and Split Data (1 Mark)

Split the dataset based on the selected attribute and create branches for its possible values.

Step 5: Repeat Recursively (1 Mark)

For each subset, repeat the process of calculating entropy and information gain using the remaining attributes.

Step 6: Stop Condition (1 Mark)

Stop when:

- All records belong to the same class, or
- No attributes are left for further splitting.

Question 6a)

Q.6	a.	Explain nearest centroid classifier and classify test instance (6, 5) using dataset in Table 6(a). Show all steps involved.	10	L3	CO3

Table 6(a)

X	Y	Class
3	1	A
5	2	A
4	3	A
7	6	B
6	7	B
8	5	B

Scheme

Working Principle / Algorithm Steps: -4 Marks

Mathematical Formula: 3 Marks

Diagram or Illustration: -2 Mark

Conclusion / Applications:-1 Mark

Example 4.3: Consider the sample data shown in Table 4.9 with two features x and y. The target classes are 'A' or 'B'. Predict the class using Nearest Centroid Classifier.

Table 4.9: Sample Data

X	Y	Class
3	1	A
5	2	A
4	3	A
7	6	B
6	7	B
8	5	B

Solution:

Step 1: Compute the mean/centroid of each class. In this example there are two classes called 'A' and 'B'.

Centroid of class 'A' = (3 + 5 + 4, 1 + 2 + 3)/3 = (12, 6)/3 = (4, 2)

Centroid of class 'B' = (7 + 6 + 8, 6 + 7 + 5)/3 = (21, 18)/3 = (7, 6)

Now given a test instance (6, 5), we can predict the class.

Step 2: Calculate the Euclidean distance between test instance (6, 5) and each of the centroid.

$$\text{Euc\_Dist}[(6,5);(4,2)] = \sqrt{(6-4)^2+(5-2)^2} = \sqrt{13} = 3.6$$

$$\text{Euc\_Dist}[(6,5);(7,6)] = \sqrt{(6-7)^2+(5-6)^2} = \sqrt{2} = 1.414$$

The test instance has smaller distance to class B. Hence, the class of this test instance is predicted as 'B'.

Question 6b)

b.	Explain concept of logistic regression with its mathematical model and application.	4	L2	CO3

Scheme: -

Definition and Concept: -1 Mark

Mathematical Model: -2 Marks

Applications: -1 Mark

Logistic Regression is a supervised machine learning algorithm used for classification problems, especially when the target variable has two classes (e.g., Yes/No, Pass/Fail, Spam/Not Spam). It predicts the probability that an instance belongs to a particular class.

Mathematical Model:

The logistic (sigmoid) function is:

$$P(Y = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n)}}$$

Where:

- $P(Y = 1)$= Probability of belonging to class 1
- β_0= Intercept
- $\beta_1, \beta_2, \ldots, \beta_n$= Model coefficients
- $x_1, x_2, \ldots, x_n$= Input features
- e= Euler's constant

The output value lies between 0 and 1 and is converted into a class label using a threshold (usually 0.5).

Applications:

- Email spam detection
- Disease diagnosis
- Credit risk assessment
- Customer churn prediction

Question 6c)

c.	Explain how the C4.5 algorithm constructs a decision tree for dataset with discrete valued attributes.	6	L2	CO3

6c)

Scheme: -

Algorithm: - 2 Marks

Construction: -3 Marks

Decision tree: - 1 Marks

Algorithm 6.3: Procedure to Construct a Decision Tree using C4.5

1. Compute Entropy_Info Eq. (6.8) for the whole training dataset based on the target attribute.
2. Compute Entropy_Info Eq. (6.9), Info_Gain Eq. (6.10), Split_Info Eq. (6.11) and Gain_Ratio Eq. (6.12) for each of the attribute in the training dataset.
3. Choose the attribute for which Gain_Ratio is maximum as the best split attribute.
4. The best split attribute is placed as the root node.
5. The root node is branched into subtrees with each subtree as an outcome of the test condition of the root node attribute. Accordingly, the training dataset is also split into subsets.
6. Recursively apply the same operation for the subset of the training set with the remaining attributes until a leaf node is derived or no more training instances are available in the subset.

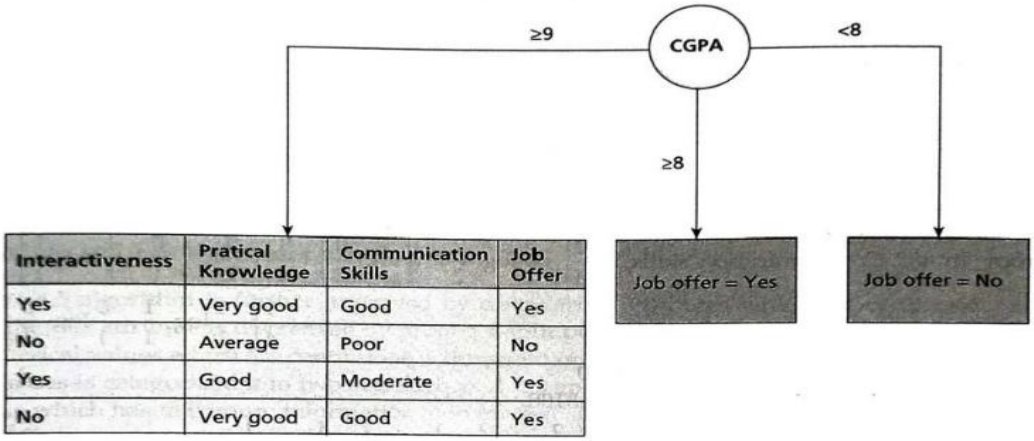
Table 6.10 shows the Gain_Ratio computed for all the attributes.

Table 6.10: Gain_Ratio

Attribute	Gain_Ratio
CGPA	**0.3658**
INTERACTIVENESS	0.0939
PRACTICAL KNOWLEDGE	0.1648
COMMUNICATION SKILLS	0.3502

Step 3: Choose the attribute for which Gain_Ratio is maximum as the best split attribute.

From Table 6.10, we can see that CGPA has highest gain ratio and it is selected as the best split attribute. We can construct the decision tree placing CGPA as the root node shown in Figure 6.5. The training dataset is split into subsets with 4 data instances.



Interactiveness	Pratical Knowledge	Communication Skills	Job Offer
Yes	Very good	Good	Yes
No	Average	Poor	No
Yes	Good	Moderate	Yes
No	Very good	Good	Yes

Figure 6.5: Decision Tree after Iteration 1

Iteration 2:

Total Samples: 4

Repeat the same process for this resultant dataset with 4 data instances.

Job Offer has 3 instances as Yes and 1 instance as No.

$$\text{Entropy\_Info(Target Class = Job Offer)} = -\frac{3}{4}\log_2\frac{3}{4} - \frac{1}{4}\log_2\frac{1}{4}$$
$$= 0.3112 + 0.5$$
$$= 0.8112$$

Interactiveness:

$$\text{Entropy\_Info}(T, \text{Interactiveness}) = \frac{2}{4}\left[-\frac{2}{2}\log_2\frac{2}{2} - \frac{0}{2}\log_2\frac{0}{2}\right] + \frac{2}{4}\left[-\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}\right]$$
$$= 0 + 0.4997$$
$$\text{Gain(Interactiveness)} = 0.8108 - 0.4997 = 0.3111$$

$$\text{Split\_Info}(T, \text{Interactiveness}) = -\frac{2}{4}\log_2\frac{2}{4} - \frac{2}{4}\log_2\frac{2}{4} = 0.5 + 0.5 = 1$$

$$\text{Gain\_Ratio(Interactiveness)} = \frac{\text{Gain(Interactiveness)}}{\text{Split\_Info}(T, \text{Interactiveness})}$$

$$= \frac{0.3112}{1} = 0.3112$$

Practical Knowledge:

$$\text{Entropy\_Info}(T, \text{Practical Knowledge}) = \frac{2}{4}\left[-\frac{2}{2}\log_2\frac{2}{2} - \frac{0}{2}\log_2\frac{0}{2}\right] + \frac{1}{4}\left[-\frac{0}{1}\log_2\frac{0}{1} - \frac{1}{1}\log_2\frac{1}{1}\right] + \frac{1}{4}\left[-\frac{1}{1}\log_2\frac{1}{1} - \frac{0}{1}\log_2\frac{0}{1}\right]$$

$$= 0$$

$$\text{Gain(Practical Knowledge)} = 0.8108$$

$$\text{Split\_Info}(T, \text{Practical Knowledge}) = -\frac{2}{4}\log_2\frac{2}{4} - \frac{1}{4}\log_2\frac{1}{4} - \frac{1}{4}\log_2\frac{1}{4} = 1.5$$

$$\text{Gain\_Ratio(Practical Knowledge)} = \frac{\text{Gain(Practical Knowledge)}}{\text{Split\_Info}(T, \text{Practical Knowledge})} = \frac{0.8108}{1.5} = 0.5408$$

Communication Skills:

$$\text{Entropy\_Info}(T, \text{Communication Skills}) = \frac{2}{4}\left[-\frac{2}{2}\log_2\frac{2}{2} - \frac{0}{2}\log_2\frac{0}{2}\right] + \frac{1}{4}\left[-\frac{0}{1}\log_2\frac{0}{1} - \frac{1}{1}\log_2\frac{1}{1}\right] + \frac{1}{4}\left[-\frac{1}{1}\log_2\frac{1}{1} - \frac{0}{1}\log_2\frac{0}{1}\right]$$

$$= 0$$

$$\text{Gain(Communication Skills)} = 0.8108$$

$$\text{Split\_Info}(T, \text{Communication Skills}) = -\frac{2}{4}\log_2\frac{2}{4} - \frac{1}{4}\log_2\frac{1}{4} - \frac{1}{4}\log_2\frac{1}{4} = 1.5$$

$$\text{Gain\_Ratio(Communication Skills)} = \frac{\text{Gain(Practical Knowledge)}}{\text{Split\_Info}(T, \text{Practical Knowledge})} = \frac{0.8108}{1.5} = 0.5408$$

Table 6.11 shows the Gain_Ratio computed for all the attributes.

Table 6.11: Gain-Ratio

Attributes	Gain_Ratio
Interactiveness	0.3112
Practical Knowledge	0.5408
Communication Skills	0.5408

Both 'Practical Knowledge' and 'Communication Skills' have the highest gain ratio. So, the best splitting attribute can either be 'Practical Knowledge' or 'Communication Skills', and therefore, the split can be based on any one of these.

Here, we split based on 'Practical Knowledge'. The final decision tree is shown in Figure 6.6.

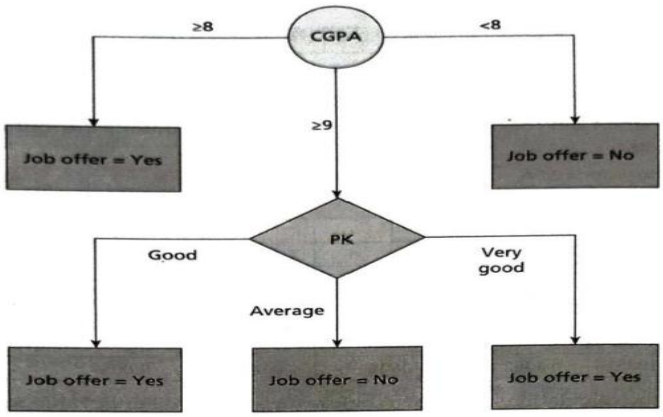


Figure 6.6: Final Decision Tree

Question 7a)

		Module – 4			
Q.7	a.	Explain Naive Bayes algorithm. Mention its applications and limitation.	10	L3	CO4

7a)

Scheme: -

Definition and Introduction: -2 Marks

Bayes Theorem and Formula: -2Marks

Working of Naïve Bayes Algorithm: -2Marks

Applications: -2 Marks

Limitations: -2Mark

- It is a supervised binary class or multi class classification algorithm that works on the principle of Bayes theorem.
- There is a family of Naïve Bayes classifiers based on a common principle.
- These algorithms classify for datasets whose features are independent and each feature is assumed to be given equal weightage.
- It particularly works for a large dataset and is very fast. It is one of the most effective and simple classification algorithms.
- This algorithm considers all features to be independent of each other even though they are individually dependent on the classified object.
- Each of the features contributes a probability value independently during classification and hence this algorithm is called as Naïve algorithm. **Some important applications of these algorithms are text classification, recommendation system and face recognition.**

Algorithm 8.1: Naïve Bayes

1. Compute the prior probability for the target class.
2. Compute Frequency matrix and likelihood Probability for each of the feature.
3. Use Bayes theorem Eq. (8.1) to calculate the probability of all hypotheses.
4. Use Maximum A Posteriori (MAP) Hypothesis, h_{MAP} Eq. (8.2) to classify the test object to the hypothesis with the highest probability.

You are performing the **above steps automatically**, like this:

Your Theory Step	What `GaussianNB` Does
Step 1 – Prior	Computes frequency of each class → estimates prior $P(h)$
Step 2 – Likelihood	Computes mean & variance of each feature for each class; assumes Gaussian distribution
Step 3 – Posterior	For new input, uses Gaussian formula to compute (P(x
Step 4 – MAP	Picks the class with the **highest posterior** score

Predict on test set

- y_pred = nb_classifier.predict(X_test)
- #It calculates posterior probabilities for each class and assigns the class with the highest one — that's your MAP classifier in action.

Applications of Naïve Bayes

1. **Email Spam Detection**
    - Classifies emails as spam or non-spam.
2. **Sentiment Analysis**
    - Determines whether reviews are positive, negative, or neutral.
3. **Document Classification**
    - Categorizes news articles, research papers, and documents.
4. **Medical Diagnosis**
    - Predicts diseases based on symptoms.
5. **Recommendation Systems**
    - Suggests products, movies, or books.
6. **Fraud Detection**
    - Identifies suspicious transactions.
7. **Text Mining and NLP**
    - Used in language processing tasks.

Limitations of Naïve Bayes

1. **Strong Independence Assumption**
    - Assumes all features are independent, which is often unrealistic.
2. **Zero Probability Problem**

- If a feature value is absent in training data, the probability becomes zero.
- Solved using **Laplace Smoothing**.

3. **Poor Performance with Correlated Features**
   - Accuracy decreases when attributes are highly dependent.
4. **Continuous Data Handling**
   - Requires assumptions (e.g., Gaussian distribution) for continuous features.
5. **Probability Estimates May Be Inaccurate**
   - Good classifier, but probability values may not reflect true likelihood.

Question 7b)

b.	Explain simple model of an artificial neuron and the structure of an Artificial Neural Network (ANN). Also describe any four activation functions.	10	L2	CO4

7b)

Scheme: -

Definition: -1 Marks

Model diagram & Explanation: -5 Marks

Activation functions: -4 Marks

- ANN structure means how neurons are arranged and connected in a network.

It defines:

- Number of layers
- Number of neurons
- How they are connected

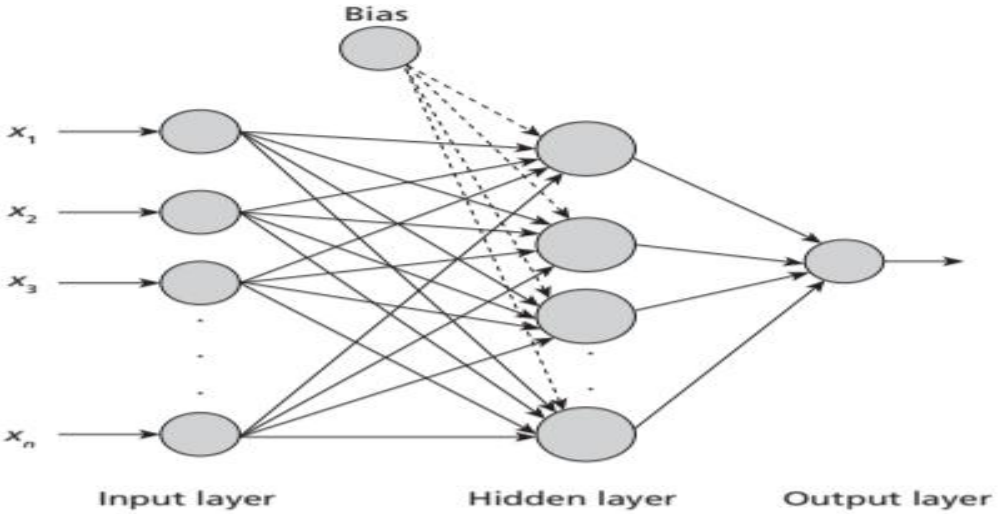


Figure 10.4: Artificial Neural Network Structure

Input Layer

- Takes input data

Example: image pixels, marks, hours studied

- No processing happens here

Hidden Layer

- Performs actual processing/calculation
- Can be one or many layers
- Uses weights, bias, and activation functions
- More hidden layers = more powerful model

Output Layer

- Gives final result
- Example:
- Pass / Fail
- Spam / Not Spam
- Disease prediction

Below are some of the activation functions used in ANNs:

1. Identity Function or Linear Function

$$f(x) = x \ \forall x \tag{10.4}$$

The value of $f(x)$ increases linearly or proportionally with the value of x. This function is useful when we do not want to apply any threshold. The output would be just the weighted sum of input values. The output value ranges between $-\infty$ and $+\infty$.

2. Binary Step Function

$$f(x) = \begin{cases} 1 & if\ f(x) \geq \theta \\ 0 & if\ f(x) < \theta \end{cases} \tag{10.5}$$

The output value is binary, i.e., 0 or 1 based on the threshold value θ. If value of $f(x)$ is greater than or equal to θ, it outputs 1 or else it outputs 0.

3. Bipolar Step Function

$$f(x) = \begin{cases} 1 & if\ f(x) \geq \theta \\ -1 & if\ f(x) < \theta \end{cases} \tag{10.6}$$

The output value is bipolar, i.e., +1 or −1 based on the threshold value θ. If value of $f(x)$ is greater than or equal to θ, it outputs +1 or else it outputs −1.

4. Sigmoidal Function or Logistic Function

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{10.7}$$

It is a widely used non-linear activation function which produces an *S*-shaped curve and the output values are in the range of 0 and 1. It has a vanishing gradient problem, i.e., no change in the prediction for very low input values and very high input values.

5. Bipolar Sigmoid Function

$$\sigma(x) = \frac{1 - e^{-x}}{1 + e^{-x}} \tag{10.8}$$

It outputs values between –1 and +1.

6. Ramp Functions

$$f(x) = \begin{cases} 1 & if\ x > 1 \\ x & if\ 0 \le x \le 1 \\ 0 & if\ x < 0 \end{cases} \tag{10.9}$$

It is a linear function whose upper and lower limits are fixed.

7. Tanh – Hyperbolic Tangent Function

The Tanh function is a scaled version of the sigmoid function which is also non-linear. It also suffers from the vanishing gradient problem. The output values range between –1 and 1.

$$\tan h(x) = \frac{2}{1 + e^{-2x}} - 1 \tag{10.10}$$

Question 8a)

		OR			
Q.8	**a.**	With a diagram of the Perceptron Model, explain the perceptron learning algorithm.	**10**	**L3**	**CO4**

8a)

Scheme: -

Diagram: -3 Marks

Explanation: -4 Marks

Algorithm: -3 Marks

1. Inputs from other neurons
2. Weights and bias
3. Net sum
4. Activation function

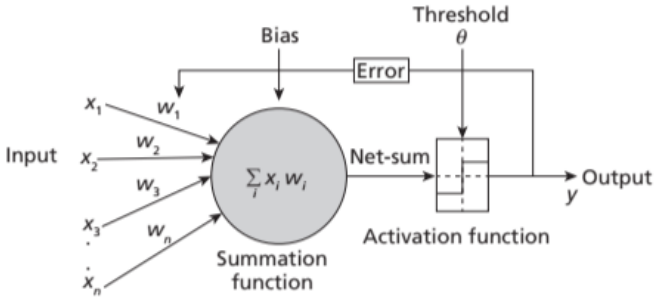


Figure 10.5: Perceptron Model

Thus, the modified neuron model receives a set of inputs $x_1, x_2, \ldots, x_n$, their associated weights $w_1, w_2, \ldots, w_n$ and a bias. The summation function 'Net-sum' Eq. (10.13) computes the weighted sum of the inputs received by the neuron.

A **perceptron** is a **linear classifier** that:

- Takes a set of input features
- Applies weights to each input
- Adds a **bias term**
- Passes the result through an **activation function** (usually a **step function** for binary classification)

$$\text{Net-sum} = \sum_{i=1}^{n} x_i w_i \tag{10.13}$$

After computing the 'Net-sum', bias value is added to it and inserted in the activation function as shown below:

$$f(x) = \text{Activation function } (\textit{Net-sum} + \textit{bias}) \tag{10.14}$$

The activation function is a binary step function which outputs a value 1 if $f(x)$ is above the threshold value θ, and a 0 if $f(x)$ is below the threshold value θ. Then, output of a neuron:

$$Y = \begin{cases} 1 & \text{if } f(x) \geq \theta \\ 0 & \text{if } f(x) < \theta \end{cases} \tag{10.15}$$

Before learning how a neural network works, let us learn about how a perceptron model works.

- In neural networks, "bias" refers to a constant value added to the output of a neuron before it's passed through the activation function.
- It acts as a threshold or offset, allowing neurons to activate even when the weighted sum of inputs is not sufficient

Simple Meaning:

- Bias → extra value added to adjust output
- Learning Rate → how fast model learns

Functions:

- Step function → gives 0 or 1
- Activation function → introduces non-linearity

Simple idea: Helps model make decisions.

- Learning rate is a hyperparameter that governs how much a machine learning model adjusts its parameters at each step of its optimization algorithm

Question 8b)

b.	Explain different types of Artificial Neural Network (ANNs) and list the major challenges associated with ANN.	10	L2	CO4

Scheme: -

Introduction to ANN: -2 Marks

Types of ANN (Any Five × 1 Mark) =5 Marks

Challenges Associated with ANN: -3 Marks

Single Layer Network

- Only input + output layer
- No hidden layer

Example: Perceptron

Used for simple problems

Multi-Layer Network (MLP)

- Has one or more hidden layers
- Fully connected network

Used for:

- Image recognition
- Speech processing
- Complex problems

Feedforward Network

- Data flows **only one direction**
  Input → Hidden → Output

No feedback or loops

Recurrent Network (RNN)

- Has **feedback connections**
- Output can go back as input

Used in:

- Text processing
- Speech recognition
- ANN structure = arrangement of neurons in layers
- 3 main layers: Input, Hidden, Output

Two common types:

- Single layer
- Multi-layer

Major Challenges of ANN:

1. Requires large training datasets.
2. High computational and memory requirements.

3. Overfitting problem.
4. Difficult hyperparameter tuning.
5. Black-box nature (lack of interpretability).
6. Long training time.
7. Sensitivity to noisy data.
8. Difficulty in choosing network architecture.

Question 9a)

Module – 5					
Q.9	**a.**	Explain different hierarchical clustering algorithms and describe linkage criteria used.	**10**	**L3**	**CO5**

9a)

Scheme:

Different type of clustering algorithm: -6 Marks

Linkage criteria: - 4 Marks

Hierarchical clustering is a method of grouping data points into clusters step by step, forming a tree-like structure called a dendrogram.

Think of it like a family tree, where small groups gradually combine into bigger groups.\

Main idea: -

It creates a nested structure of clusters

Clusters are formed at different levels (small → big)

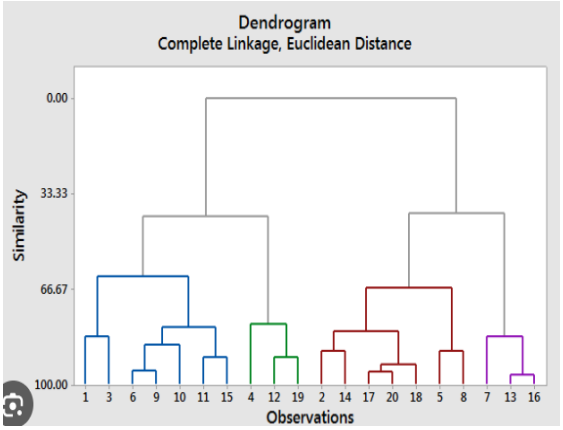
Relationships between data points are shown hierarchically

Hierarchical methods produce a nested partition of objects with hierarchical relationships among objects. Often, the hierarchy relationship is shown in the form of a dendrogram.

Hierarchical methods include categories, agglomerative methods and divisive methods.

In agglomerative methods, initially all individual samples are considered as a cluster, that is, a cluster with a single element

Then, they are merged and the process is continued to get a single cluster.



- Divisive methods use another kind of philosophy, **where a single cluster of all samples of the dataset taken initially is chosen and then partitioned.**
- This **partition process is continued until the cluster is split into smaller clusters.**
- **Agglomerative methods merge clusters to reduce the number of clusters.**
- This is repeated each time while merging two closest clusters to get a single cluster.

Algorithm 13.1: Agglomerative Clustering

1. Place each N sample or data instance into a separate cluster. So, initially N clusters are available.
2. Repeat the following steps until a single cluster is formed:
   (a) Determine two most similar clusters.
   (b) Merge the two clusters into a single cluster reducing the number of clusters as N–1.
3. Choose resultant clusters of step 2 as result.

All the clusters that are produced by hierarchical algorithms have equal diameters. The main disadvantage of this approach is that once the cluster is formed, it is an irreversible decision.

Question 9b)

b.	Explain density based clustering with DBSCAN. Mention its advantages.	**10**	**L2**	**CO5**

9b)

Scheme: -

DBSCAN Algorithm: - 6 Marks

Advantages: - 4 Marks

- **DBSCAN (Density-Based Spatial Clustering of Applications with Noise)** is a clustering algorithm that groups data points based on **density** rather than distance alone.

Key Idea

- **Instead of forming clusters by shape (like circles in k-means), DBSCAN identifies regions where many points are closely packed together.**
- **Areas with high density → clusters**
- **Areas with low density → noise (outliers)**

Density means:

- A region where the number of points is greater than a specified threshold within a given distance.

In DBSCAN, two important parameters define this:

- **ε (epsilon)** → radius (distance to search neighbors)
- **MinPts** → minimum number of points required to form a dense region

For each point, check how many neighbors are within distance ε.

- If neighbors ≥ MinPts → it becomes a **core point** (start of a cluster).
- Points close to core points → added to the cluster (**border points**).

Points that do not belong to any cluster → **noise (outliers)**.

Algorithm 13.4: DBSCAN

Step 1: Randomly select a point p. Compute distance between p and all other points.

Step 2: Find all points from p with respect to its neighbourhood and check whether it has minimum number of points m. If so, it is marked as a core point.

Step 3: If it is a core point, then a new cluster is formed, or existing cluster is enlarged.

Step 4: If it is a border point, then the algorithm moves to the next point and marks it as visited.

Step 5: If it is a noise point, they are removed.

Step 6: Merge the clusters if it is mergeable, $dist\ (c_i, c_j) < \varepsilon$.

Step 7: Repeat the process 3–6 till all points are processed.

Advantages of DBSCAN:

- Does not require the number of clusters in advance.
- Detects clusters of arbitrary shapes.
- Handles noise and outliers effectively.
- Robust to outliers.
- Works well on large datasets.
- Less sensitive to initialization.
- Suitable for spatial and real-world data clustering.

Question 10a)

		OR			
Q.10	a.	Explain Reinforcement Learning (RL), its characteristics and challenges.	10	L2	CO5

10a)

Scheme: -

Reinforcement Learning: -5 Marks

Characteristics: -3 Marks

Challenges: -2 Marks

- Reinforcement learning (RL) mimics human beings and is a branch of machine learning. Humans observe the environment through senses such as eye and ear. The inputs are processed by brain.
- The brain then suggests the actions and acts voluntarily or involuntarily. In humans, learning happens in a real world called environment.
- Thus, reinforcement learning is a mathematical framework for learning.
- There are two types of reinforcement learning – **positive and negative**.
- **Positive reinforcement learning** is a recurrence of behaviour due to positive rewards. Rewards increase strength and the frequency of a specific behaviour. This encourages to execute similar actions that yield maximum reward.
- Similarly, in **negative reinforcement learning**, negative rewards are used as a deterrent to weaken the behaviour and to avoid it.
- In a **maze game**, there may a danger spot that may lead to loss. Negative rewards can be designed for such spots so that the agent does not visit that spot.
- Positive and negative rewards are simulated in reinforcement learning, say +10 for positive reward and -10 for some danger or negative reward.
- ***Reinforcement learning is an example of semi-supervised learning technique and is used to model sequential decision-making process.***

Characteristics of Reinforcement Learning

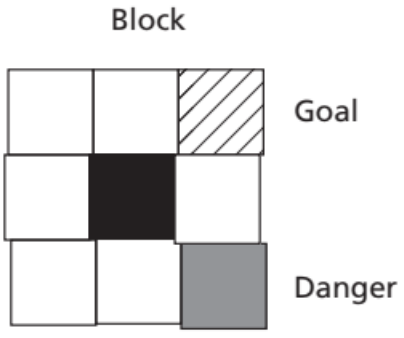


Figure 14.3: A Grid Game

Sequential decision making – Consider the Figure 14.3. It can be seen the path from start to goal is not done in one step. It is a sequence of decisions that leads to the goal. One wrong move may result in a failure. This is the main characteristic of reinforcement learning.

2. **Delayed feedback** – Often, rewards are not immediate. One must spend many moves to get final success or failure. Feedback in terms of reward is often delayed.

3. The agent actions are interdependent as any action affects the subsequent actions. For example, one wrong move of an agent may lead to failure.

4. **Time related** – All actions are associated with time stamps inherently as all actions are ordered as per the timeline inherently.

Challenges of Reinforcement Learning

1. **Reward design** is a big challenge as in many games, as determining the rewards and its value is a challenge.

2. **Absence of a model is a challenge** – Games like chess have fixed board and rules. But, many games do not have any fixed environment or rules. There is no underlying model as well. So, simulation must be done to gather experience.

3. **Partial observability of states** – Many states are fully observable. Imagine a scenario in a weather forecasting where the uncertainty or partial observability exists as complete information about the state is simply not available.

4. **Time consuming operations** – More state spaces and possible actions may complicate the scenarios, resulting in more time consumption.

5. **Complexity** – Many games like GO are complicated with much larger board configuration and many possibilities of actions. So, labelled data is simply not available. This adds more complexity to the design of reinforcement algorithms.

Question 10 b)

b.	Explain the Q-learning algorithm with its steps and formulas.	**10**	**L2**	**CO5**

10b)

Scheme: -

Q Learning Algorithm: - 6 Marks

Formula: - 4 Marks

- Q-Learning is an off-policy method because Q is updated based on policies that are implicit to Q and is better guaranteed for maximum returns. Q indicates Quality.
- What is Q-value? Q (s, a) is a numerical value assigned to a state-action pair. It means a value of the action that is performed in state ‘s’. Here, the main objective is to find Q-Value?
- Q-value is nothing but the immediate reward and other rewards that are yet to come, known as total return reward.
- To compute the total return reward, these information are necessary.
- 1. Starting state
- 2. Action

- 3. Reward

4. New state

Initially, a table called *Q*-table is constructed and filled with initial random values. *Q*-learning involves two policies - learning policy and update policy. *Q*-learning is done by methods like greedy, softmax or softmax plus.

The agent's next move will be the action where rewards are high. Alternatively, it would be the cell that has the highest *Q*-value. As the moves are determined by *Q*-values, the computation and updation of *Q*-values are important for *Q*-learning. Then comes the *Q*-learning update policy. The updation of *Q*-values is done using Temporal-Differencing method that is mentioned in section 14.8.2. This updation is made by blending the new and old values. Blending is done by α. When α is zero, the value does not change at all. When α is one, the next value replaces the old value. Here, α is known as the learning rate. The discount factor (γ) is also used for update. Blending is done using temporal difference learning.

The updation procedure of *Q*-value is given as follows:

1. Perform any random action on state s_t
2. Get a new state, s_{t+1}
3. Get the award $r(s_t, a_t)$

The temporal difference at time 't' is done using the update:

$$TD_t(s_t, a_t) = r(s_t, a_t) + \gamma \max_a(Q(s_{t+1}, a)) - Q(s_t, a_t) \quad (14.23)$$

Here, $r(s_t, a_t)$ is the reward obtained by performing action a_t and $Q(s_{t+1}, a)$ is the estimate of the best action at state s_{t+1} i.e., s' and $Q(s_t, a_t)$ is the *Q*-value of the action a_t at state s_t. Here, the best action performed at future states discounted by discount factor (γ) is $\gamma \max_a(Q(s_{t+1}, a)) - Q(s_t, a_t)$.

If TD is high, then it is a 'surprise factor' and denotes the highest reward. If TD is less, it represents the 'frustration' factor and denotes the lesser reward. In other words, TD is a sort of 'reward'. It is initially high and slowly gets minimized as it reaches the end of the training, that is, when it reaches the final goal.

The update is carried out using the Temporal difference learning (TD) and Bellman equation. The Bellman equation that is used to update the value is given as follows:

$$Q_t(s_t, a_t) = Q_t(s_t, a_t) + \alpha TD_t(s_t, a_t) \quad (14.24)$$

The *Q*-Learning algorithm is given as follows:

Algorithm 14.7: *Q*-Learning

1. Set $Q(s_t, a) = 0$ where s_t is the terminal node in an episode.
2. For all states s and action a, set *Q* Value = 0.
3. Repeat.
4. Select s_t randomly.
5. Choose action a_t from *Q* using ε- greedy.
6. Perform action a_t such that $r(s_t, a_t) > 0$ and reach the next state.
7. Update the action-value function using TD using Eq. (14.24) and Bellman equation:

$$Q_t(s_t, a_t) = Q_{t-1}(s_t, a_t) + \alpha \times TD_t(s_t, a_t) \quad (14.25)$$

Until the terminal state s is reached.

Here, α is the learning rate. It is between 0 and 1. If α is zero, then nothing is updated and there is no learning at all. When setting it to a higher value such as 0.9, the learning happens fast. γ is the discount factor. It is in the range $0 - 1$. It should be less so that the algorithm can converge.

The inference is simple, the best action is the maximum of *Q* function as follows:

$$a_t = \arg\max_a(Q(s_t, a))$$